

REMARKS

Claims 55 – 85 are currently pending in this response. Claims 55 – 85 are rejected. Claims 55, 80 and 81 are amended in this response.

Claim Rejections – 35 USC 102

The Examiner rejected claims 55 – 85 under 35 USC 102(e), as being anticipated by Helgeson et al. US Patent No. 6643652.

Claim 55 has now been amended to specify that the object is a software object hosted on a server, and is respectfully believed to overcome the rejection to Hegelson. The Examiner on page 2 of his Office Action still appears to be interpreting the interfacing server as reading onto the object of the claim, even though in applicant's view the claim already said that the servers hosted object. The present amendment is therefore a mere clarification to avoid all doubt and should not be understood as a substantive limitation of the claim.

Thus claim 55 has been amended to specify *a software object that is hosted on a server*. A *software object* is a term *well known in the art* of computer programming and refers to a self-contained unit –generally having methods, attributes and data so that it acts as an autonomous unit within a program. Such software objects are commonly used in the programming style known as Object Orientated Programming or OOP.

A hosting server is *not* hosted, it does not make any sense to say that a server is hosted on a hosting server, and Hegelson certainly does not say that. Hegelson therefore cannot read onto a software object *hosted on a server*.

It is accepted that Hegelson teaches generating an XSP page (paragraph 0691) and transforming it into a Java object. The object can be initialized (paragraph 0696).

It is accepted that software objects have attributes.

However there is no teaching that the XSP page Java *object* of Hegelson comprises:

a first identity arrangement for holding said first unique identity indicating one of said plurality of hosting servers or provider of said object, and

a second identity arrangement for holding respective second unique identities of specific remote entities establishing a relationship with said object via a network through respective remote terminal devices,

This is because the model page discussed by Hegelson is located on a *particular* hosting server with which it is *always identified* and simply interacts with anyone accessing the page, without enabling the accessor to pass the model page to a 3rd-party who does not know the model page's hosting server. In any case, the model pages only contain data, so there is no need to access the hosting server to get the object's behavior.. There is thus no need for specific attributes identifying either the hosting server or the remote entity.

Irrespective of the need, there is *no evidence* that Hegelson supplies *such attributes in combination within a software object hosted on a server*.

Thus claim 55 is believed to be novel and inventive over Helgeson.

The same features of the object are defined in claims 80 and 81 and these claims are therefore believed to be novel and inventive for the same reasons.

Examiner's Arguments

Although the Examiner does point to Figure 4, items 419, 421 and 423 as being identity arrangements, these in fact point to servers, so this point is believed to be rendered moot by the present amendments.

The Examiner additionally points to paragraph 519 lines 1 – 10, but it is respectfully pointed out that this paragraph only has 8 lines. In addition it simply contains a general reference to the prior art being tightly bound to application products etc, and offers no solution, so it is difficult to see any identity arrangement therein.

As regards the first identity itself the Examiner makes the remark “that's XML protocol”, without any supporting reference. Applicant is not aware of a protocol within XML for *mandating an identification of the hosting server*.

Again as no software object hosted on a server is indicated the point is rendered moot by the present amendments. Indeed XML – as a mark-up language rather than a programming language per se, could not support a software object. It could merely provide data for the content of the object.

For the second identity arrangement the Examiner points to paragraph 526 lines 1 - 3. This paragraph however simply talks about a web content generation

engine and there is no mention of a remote user or of any arrangement for storing the identity of the remote user. Applicant wonders if the Examiner possibly intended a different paragraph. An identity of a remote user is hardly web content which would be generated by a web content generation engine.

The Examiner points to paragraph 215 lines 10 – 14 as teaching the remote user. However lines 10 – 14 merely teach a combination of servers and hardware boxes. There are users referred to in paragraph 215, since web clients" are users. But the paragraph doesn't mention any way to *identify* them – that is, if the same set of clients repeatedly interact with the same server, there is *no way to identify which client is which*, and again, applicant is somewhat confused at the rejection.

Examiner is expected to interpret the claims in light of the broadest *reasonable* interpretation.

In any event the point is moot as there is no mention of a software object hosted on a server.

Examiner finds the feature of allowing more than one user to approach the data simultaneously, from paragraphs 382 and 385 which discuss– persistence of data. Indeed paragraph 385 says the exact opposite.

“concurrent access for session beans is *not meaningful* since by definition an instance of a stateful session can be used by only one client, and stateless sessions do not maintain any data that needs to be shared”,(Hegelson paragraph 385 lines 7 – 10 – emphasis added).

In this paragraph, session beans and data objects are two different things – session beans are a way to manage access to the *same* data objects by multiple users, i.e. data objects that can be accessed by multiple users. Each user has his own session bean and all the users then jointly access the same object. However there is no knowledge at the object of the individual users.

Paragraph 364 is just a single word title – relationships, and paragraph 365 is about how entities can inherit from a class, and has nothing to do with users, remote users or multiple remote users.

Conclusion

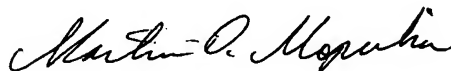
All the matters raised by the Examiner are believed to have been dealt with.

No new matter is added by the present amendments.

Claims 55, 80 and 81 are believed to be novel and inventive over Hegelson, who does not teach or hint at a *software object hosted on a server*, combining together *two identity arrangements* for providing to the software object *two identification attributes* being a *hosting server identification* and a *remote user identification*.

All the matters raised by the Examiner have been dealt with and allowance of the application is respectfully awaited.

Respectfully submitted,



Martin D. Moynihan
Registration No. 40,338

Date: July 28, 2008

Enclosures:

- Petition for Extension (1 Month)
- Request for Continued Examination (RCE)